

Amendment to the Claims

What is claimed is:

1. (Original) A failover method for a computer system having at least three nodes operating as a cluster, said method comprising the steps of:

following the failure of one of said nodes, determining the weight of at least two surviving nodes;

determining which of said at least two surviving nodes has the lowest weight;

assigning applications running on said failed node to said surviving node having the lowest determined weight.

2. (Original) The method according to claim 1, wherein in said weight determining step, the weight of every one of said surviving nodes is determined.

3. (Original) The method according to claim 2, wherein said weight is determined by evaluating available resources of said node.

4. (Original) The method according to claim 3, said evaluating step further comprising the steps of:

examining at least one performance indicator associated with said node; and

using a predetermined method to determine from said at least one performance indicator said weight of said node.

5. (Original) The method according to claim 4, wherein said at least one performance indicator is an indicator of current CPU utilization of that node.

6. (Original) The method according to claim 3, said assessing step further comprising the steps of examining at least two performance indicators associated with said node; and using a predetermined mathematical formula to calculate from said performance indicators said weight of said node.

7. (Original) The method according to claim 6, wherein said at least two performance indicators include an indicator of CPU utilization of that node, and an indicator of the amount of memory currently being used by that node.

8. (Original) A failover method for a computer system having at least three nodes operating as a cluster, said method comprising the steps of:

determining the amount of resources needed by applications running on one of said nodes;

following failure of said one node, for each of surviving nodes $n = 1$ to N until a failover node is assigned:

determining a weight of surviving node n ;

determining from said weight of said surviving node n whether said surviving node n has available resources greater than that determined to be needed by said failed node; and

if said surviving node n is determined to have sufficient available resources, then assigning node n as said failover node, and failing over applications running on said failed node to said failover node, or if said surviving node n is determined not to have sufficient available resources, then $n=n+1$.

9. (Cancelled).

10. (Original) A failover method for a computer system having at least three nodes operating as a cluster, said method comprising the steps of:

determining a weight of each of said at least three nodes;

ordering said at least three nodes according to their respective increasing weights from lowest to highest;

creating a queue containing said ordered nodes, wherein said first node in said queue has the lowest weight; and

following failure of one of said at least three nodes, assigning said first surviving node in said queue as a failover node, and failing over applications running on said failed node to said failover node.

11. (Withdrawn) A failover method for a computer system having at least three nodes operating as a cluster, said method comprising the steps of:

following failure of one of said at least three nodes, determining the order in which surviving nodes joined said cluster;

assigning a failover node according to the order in which said surviving nodes joined said cluster; and

failing over all applications running on said failed node to said failover node.

12. (Withdrawn) The failover method according to claim 11, further comprising the steps of:

determining a first surviving node to join said cluster, and

assigning said first joined surviving node as said failover node.

13. (Withdrawn) The failover method according to claim 11, further comprising the steps of:

determining a last surviving node to join said cluster; and

assigning said last joined surviving node as said failover node.

14. (Withdrawn) A failover method for a computer system having at least three nodes operating as a cluster, said method comprising the steps of:

detecting failure of one of said at least three nodes;

determining a time of failure of said one node;

assigning a failover node taking into consideration said determined time; and

assigning applications running on said failed node to said failover node.

15. (Withdrawn) The method according to claim 14, further comprising the steps of:
for at least one node in said cluster, determining a time period during which said node is heavily utilized, and
preventing said at least one node from being assigned as a failover node during said time period during which it is heavily utilized.

16. (Withdrawn) The method according to claim 15, further comprising the steps of:
for at least said failed node, determining in advance of failure a time during which at least one application running on said failed node is heavily utilized;
following failure of said node, if failure occurs during said determined time period, then assigning a failover node for said at least one application first.

17-19. (Cancelled).

20. (Withdrawn) A cluster computer system comprising:
at least three nodes, wherein said at least three nodes are computer systems operating as a cluster, wherein said cluster computer system is capable of determining a time of failure of one of said at least three nodes, and implementing a failover policy in which, following failure of said one node, a failover node is selected based in part on said determined time.

21. (Withdrawn) The cluster computer system according to claim 20, wherein said cluster computer system is further capable of determining a time period during which at least one

of said nodes is heavily utilized, and preventing said at least one node from being assigned as a failover node during said determined time period.

22. (Withdrawn) The cluster computer system according to claim 21, wherein said cluster computer system is further capable of determining a time during which at least one application on said failed node is heavily utilized, and upon failure of said failed node, if said failure occurs during said determined time, then failing over said at least one application first.

23. (Amended) A failover method for a computer system having at least three nodes operating as a cluster, said method comprising the steps of:

detecting the failure of one of said nodes;

determining the weight of at least two surviving nodes, wherein for each of said at least two surviving nodes, said weight is determined by evaluating at least a performance indicator indicating current CPU utilization of the surviving node;

determining which of said at least two surviving nodes has the lowest weight; and

assigning applications running on said failed node to said node determined to have said lowest weight.

24. (Withdrawn) A failover method for a computer system having at least three nodes operating as a cluster, said method comprising the steps of:

determining a time period during which each of said at least three nodes are heavily utilized;

detecting failure of one of said at least three nodes;

determining a time of failure of said one node;

preventing failover of applications running on said one node to other of said at least three nodes if said determined time period for said other node includes said determined time of failure.

25. (New) A failover method for a computer system having at least three nodes operating as a cluster, said method comprising the steps of:

for each node, determining the amount of resources needed by applications running on the node;

following failure of a node,

selecting a node from among the group of surviving nodes;

determining a weight of the selected surviving node;

evaluating on the basis of the determined weight of the selected surviving node whether the selected surviving node has more available resources than the resources needed by said failed node;

if the selected surviving node n has more available resources than the resources needed by said failed node, assigning the selected surviving node as the designated failover node and migrating any applications of the failed node to the designated failover node;
and

if the selected surviving node does not have more available resources than the resources needed by said failed node, selecting another surviving node from the group of surviving nodes.

Respectfully submitted,



Roger Fulghum

Registration No. 39,678

Baker Botts L.L.P.
910 Louisiana
One Shell Plaza
Houston, Texas 77002-4995
(713) 229-1707

Baker Botts Docket Number: 016295.0672

Date: November 2, 2004